

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-273272

(43)Date of publication of application : 05.10.2001

(51)Int.Cl.

G06F 15/16
G06F 11/00
G06F 11/16
G06F 12/00

(21)Application number : 2000-084541

(71)Applicant : NTT COMWARE CORP

(22)Date of filing : 24.03.2000

(72)Inventor : OMI TAKAHARU

FUJINAMI KAORI

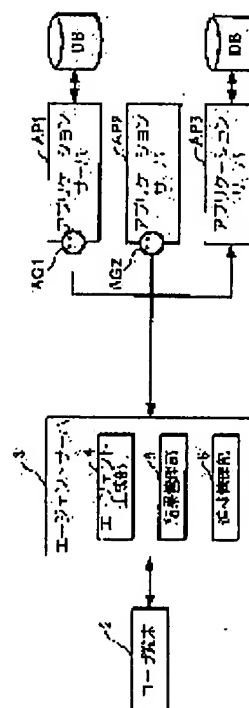
NAGAOKA TORU

(54) CONTROL METHOD FOR AGENT SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a control method for agent system capable of continuing subsequent processing while using information, which is acquired by an agent before the occurrence of a fault such as the extinction of the agent, even when the fault occurs.

SOLUTION: When the processing result in an application server AP1 of a moving destination is completely acquired, the agent is moved to an application server AP1 of the next moving destination as an agent AG2 and makes a copy agent AG1, which is the copy of the agent itself, having acquired the processing result at that point of time, wait in the relevant application server AP1.



LEGAL STATUS

[Date of request for examination] 26.03.2001

[Date of sending the examiner's decision of] 10.02.2004

rejection]

[Kind of final disposal of application other than
the examiner's decision of rejection or
application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's
decision of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] According to a demand of a user, an agent server a network top by generating a movable agent In the control approach of an agent system of performing processing corresponding to a demand of said user in two or more application servers connected to said network through said agent If acquisition of the processing result in said application server of a migration place is completed, said agent Create the self duplicate at the time and the application server concerned is made to stand by as a copy agent. The agent concerned The control approach of the agent system characterized by moving to said application server which is the following migration place, or said agent server.

[Claim 2] It is the control approach of the agent system characterized by deleting a copy agent including the copy agent created at least among said copy agents who are living whenever said agent moves at the very beginning after said copy agent becomes the number defined beforehand in the control approach of an agent system according to claim 1.

[Claim 3] The control approach of the agent system characterized by the newest copy agent deleting said copy agent who should delete among said living copy agents in the control approach of an agent system according to claim 2.

[Claim 4] It is the control approach of the agent system characterized by to perform predetermined processing that either of said living copy agents should continue processing of an agent that said disappearance or failure was detected when it sets to the control approach of an agent system according to claim 1 , said agent server has the status-control section which manages said agent's current location and current condition and said status-control section detects said agent's disappearance or failure .

[Claim 5] It is the control approach of the agent system characterized by performing predetermined processing that processing of an agent in which said disappearance or failure was detected should be continued when either of said copy agents detects said agent's disappearance or failure in the control approach of an agent system according to claim 1.

[Claim 6] Said predetermined processing is the control approach of the agent system characterized by being the processing of which the lock of said application server which the agent by whom said disappearance or failure was detected in the control approach of an agent system according to claim 4 or 5 locked is canceled.

[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the control approach of an agent system.

[0002]

[Description of the Prior Art] The agent system who makes it move to an application server directly, and performs the agent who is a program for making various processings perform in recent years is proposed. It connects by the communication line and, as for this agent system, the agent server which notifies the information acquired by the agent to a user terminal with a user terminal while generating an agent, and two or more application servers which perform processing corresponding to the demand from an agent, and return a processing result to an agent are constituted, respectively.

[0003] That is, in an agent system, if there is a demand from a user through a user terminal, the agent for acquiring the result which corresponds to a demand of a user by the agent server will be generated, and it will move to the application server which this agent should move first. And if an agent makes the processing corresponding to a demand of a user perform to the application server which moved and acquisition of the processing result is completed, he will move to the agent server of the following migration place. Thus, an agent does sequential migration at two or more application servers, and if the result corresponding to a demand of a user is acquired, return and an acquisition result will be notified to an agent server at an agent server. Thereby, an agent server can notify a user of the result which the user demanded through a user terminal. Thereby, since an agent system does not always need to connect the circuit of the server of a client side, and an application server like the conventional client/server system, it can acquire the information on desired from an application server with small traffic.

[0004]

[Problem(s) to be Solved by the Invention] By the way, in the agent system of such a distributed application environment, when failures, such as processing failure, occur in the application server in which an agent exists, there are some which the agent acquired the fault information, processed roll forward who returns to the roll back which returns to the first application server, or the last migration place, and made processing continuable. However, since it was necessary for an agent to acquire fault information, this approach was not taken into consideration when a failure with which the agent itself, such as a system down of an application server, disappears occurred. For this reason, when the agent itself disappeared, even if processing was able to be interrupted or it has detected that the agent disappeared even if, after all, the agent newly needed to be generated and it needed to process anew.

[0005] Then, the purpose of this invention aims at offering the control approach of the agent system which can continue the processing which will continue using the information which the agent acquired by the time of failure generating, even when a failure with which an agent disappears occurs.

[0006]

[Means for Solving the Problem] In order to solve the above-mentioned technical problem, a configuration according to claim 1 According to a demand of a user, an agent server a network top by generating a movable agent In the control approach of an agent system of performing processing

corresponding to a demand of said user in two or more application servers connected to said network through said agent. If acquisition of the processing result in said application server of a migration place is completed, said agent creates the self duplicate at the time, the application server concerned is made to stand by as a copy agent, and the agent concerned is characterized by moving to said application server which is the following migration place, or said agent server.

[0007] In the control approach of an agent system according to claim 1, the configuration according to claim 2 is characterized by deleting a copy agent including the copy agent created at least among said copy agents who are living whenever said agent moves at the very beginning, after said copy agent becomes the number defined beforehand.

[0008] The configuration according to claim 3 is characterized by the newest copy agent deleting said copy agent who should delete among said living copy agents in the control approach of an agent system according to claim 2.

[0009] When a configuration according to claim 4 is set to the control approach of an agent system according to claim 1, said agent server has the status control section which manages said agent's current location and current condition and said status control section detects said agent's disappearance or failure, either of said living copy agents is characterized by to perform predetermined processing that processing of an agent in which said disappearance or failure was detected should continue.

[0010] In the control approach of an agent system according to claim 1, when either of said copy agents detects said agent's disappearance or failure, as for the configuration according to claim 5, it is characterized by said disappearance or failure performing predetermined processing that processing of the detected agent should be continued.

[0011] The configuration according to claim 6 is characterized by said predetermined processing being processing of which the lock of said application server which the agent by whom said disappearance or failure was detected locked is canceled in the control approach of an agent system according to claim 4 or 5.

[0012]

[Embodiment of the Invention] Hereafter, the operation gestalt of this invention is explained, referring to a drawing suitably.

[0013] (1) Operation gestalt (1-1) The block diagram 1 of an operation gestalt is a block diagram showing the agent system concerning the operation gestalt of this invention. It connects by the communication line and, as for this agent system 1, the user terminal 2 which is a terminal unit for a user to use this system, the agent server 3 which acquires the result which corresponds to a demand from a user terminal 2, and is returned to a user terminal 2, and two or more application servers AP1-AP3 which perform application of performing processing to the demand from Agent AG, and returning a processing result to Agent AG are constituted, respectively.

[0014] The agent server 3 consists of the Management Department 5 and the status-control section 6 a result with the agent generation section 4. The agent generation section 4 analyzes the demand of a user terminal 2, reads the predetermined source program memorized by the memory which is not illustrated according to an analysis result, generates a predetermined program, and generates the agent AG who adds and mentions migration place information later to this program. Each agent AG manages the processing result acquired from application servers AP1-AP3, and the result Management Department 5 transmits the information corresponding to a demand of a user to a user terminal 2. When each agent's AG current location and the condition of survival are supervised and Agent AG (the copy agent corresponding to this agent AG that mentions later is included) judges that continuation of processing is impossible, the status-control section 6 makes this agent AG create again, and processing is made to continue or it performs termination processing.

[0015] As application servers AP1-AP3 were mentioned above, it is the server which performs application which performs predetermined processing to the demand from Agent AG, for example, application servers AP1 and AP3 are servers which search the data of the request [**** / updating the contents of the database DB] out of Database DB according to a demand of Agent AG, and return a retrieval result to Agent AG in drawing 1.

[0016] Agent AG is a program which moves itself between the agent server 3 and application servers AP1-AP3 and between application servers AP [AP1-] 3 that the result which receives a demand of a user should be acquired. This agent AG is the program which can perform recovery at the time of the failure which opts for subsequent actuation independently, when it made the application server of a migration place perform predetermined processing, and it not only notifies the current position and a condition to the status-control section 6 of an application server 3, but it acquires a processing result or failures, such as transaction management and processing failure in the application server of a migration place, occur, as mentioned above.

[0017] That is, when Agent AG detects failures, such as processing failure of the application server 3 of a migration place, Agent AG can move to other application servers which can process the processing which the application server which its failure besides [which performs a roll back and roll forward] an approach generated should perform, or it can move to the agent server 3, and the optimal approach can be independently chosen out of the various approaches of notifying a result the middle. Therefore, in this agent system 1, when Agent AG detects the failure of the application servers AP1-AP3 of a migration place, Agent AG can cope with that failure independently, and can continue processing.

[0018] Moreover, in this agent system 1, when Agent AG moves between application servers AP [AP1-] 3, the copy agent who is that agent's AG duplicate stands by to the application server of the migration place in front of one. That is, if acquisition of a processing result is completed in the application server AP 1 whose agent AG1 is the first migration place as shown in drawing 2 as an agent's moving method is shown in drawing 2 - drawing 4 for example, the duplicate of the agent AG1 who acquired the processing result will be created. And as shown in drawing 3 , it moves to the application server AP 2 the duplicate of whose is the following migration place as an agent AG2, and a reproducing agency stands by to an application server AP 1 as a copy agent AG1.

[0019] Next, while the duplicate of the agent AG2 who acquired the processing result till then will move to the application server AP 3 which is the following migration place as an agent AG3 like the case where it is a **** as shown in drawing 4 if acquisition of a processing result completes the agent AG2 who moved to the application server AP 2 in an application server AP 2, the copy agent AG2 stands by to an application server AP 2. At this time, the copy agent AG1 who stands by to an application server AP 1 by the copy agent AG2 who stands by to an application server AP 2 is deleted, and the copy agent who acquired the last processing result only to the application server of agent's who moved at the end migration origin stands by.

[0020] Therefore, when the agent's itself disappearance is detected by the copy agent by the down of the application server in which Agent AG exists now etc., a roll back, roll forward, etc. who mentioned above instead of the agent AG to whom the copy agent disappeared can be processed. That is, since a copy agent has the function to cope with a failure independently like the usual agent while having the processing result acquired by the time the agent who disappeared disappeared, he can cope with a failure quickly like the case where an agent detects the failure of an application server.

[0021] Even when the agent itself disappears in this agent system 1 by this by the down of the application server in which the agent exists etc., processing can be continued by the copy agent who exists in the application server of the migration origin of the application server which the agent who disappeared moved at the end. Moreover, since this copy agent has the processing result acquired by the time the agent who disappeared moved from that application server that stands by, he can overlap, does not need to perform processing which the agent who disappeared already performed, and can continue processing quickly.

[0022] In addition, in this agent system 1, if the status-control section 6 of the agent server 3 detects these disappearances when a failure occurs in two or more application servers and an agent and all the copy agents disappear, processing is continuable with the agent AG newly generated by the agent generation section 4. Moreover, a failure does not occur to the agent system 1, but when the result to which the agent server 3 corresponds to a demand of a user 2 through Agent AG is able to be acquired, a copy agent is deleted by control of the agent server 3.

[0023] (1-2) The agent system 1 concerning the effectiveness book operation gestalt of an operation

gestalt If Agent AG completes acquisition of the processing result in the application server of a migration place By making the copy agent who is the duplicate of the agent who acquired the processing result stand by to the application server Even when a failure with which the agent AG who moved to the following migration place disappears occurs, the processing which will continue using the information which the agent acquired by the time of failure generating can be continued.

[0024] (2) Modification (2-1) Although the case where an agent made a copy agent stand by only to the application server of the migration origin of the application server which is recognizing current existence was described, this invention may make a copy agent stand by to two or more application servers in the 1st modification above-mentioned operation gestalt, respectively that what is necessary is just to make a copy agent stand by to application servers not only this but other than the application server the agent is recognizing [the application server] current existence in short. In this case, even when a failure occurs in two or more application servers and an agent and one of copy agents disappears, processing can be efficiently continued by the copy agent created at the end among the copy agents who stand by to a normal application server.

[0025] Moreover, although it is desirable to delete the copy agent created among the copy agents who are living whenever a copy agent is newly created at the very beginning after the number of copy agents turns into a number defined beforehand, the number of the copy agents who you may determine [agents] the copy agent who deletes based on the contents of processing of the application server which stands by etc., and make it stand by may change dynamically. In addition, a copy agent may be made to stand by to all the application servers that the agent moved, without deleting a copy agent.

[0026] (2-2) In the 2nd modification above-mentioned operation gestalt, although the case where the processing which the status-control section 5 of the agent server 3 which detected an agent's disappearance follows was directed to a copy agent was described, as for this invention, a copy agent may detect a direct agent's disappearance by the agent communication link between not only this but an agent, and a copy agent.

[0027] (2-3) Only when an agent disappears, in the 3rd modification above-mentioned operation gestalt, the case where processing of an agent in_which the copy agent created at the end disappeared is continued described, but this invention may perform the processing which the copy agent created at the end instead of the agent whom the failure generated follows, when a failure occurs not only to this but to the agent itself and it becomes processing continuation impossible.

[0028] (2-4) In the 4th modification and an above-mentioned operation gestalt, in order that an application server while performing processing which a certain agent demands may prevent performing processing by other agents, this invention may be applied to the agent system which performs lock processing an agent makes [processing] the application server under processing activation lock [processing]. In this agent system, since the application server which the agent who disappeared locked remains locked when the agent who moved to the application server disappears according to a failure, the copy agent created at the end may perform discharge of this lock.

[0029] (2-5) In the 5th modification above-mentioned operation gestalt, although the case where this invention was applied to the agent system of the distributed application environment where one user terminal was connected to the agent server was described, this invention can apply this invention to the agent system of the distributed application environment where two or more user terminals were connected not only to this but to the agent server, and the agent system of a distributed application environment with two or more agent servers, widely.

[0030]

[Effect of the Invention] As mentioned above, the agent system of this invention can continue the processing which will continue using the information which the agent acquired by the time of failure generating by the copy agent who stands by to the application server which the failure has not generated, even when a failure with which an agent disappears occurs.

[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the block diagram of the agent system concerning the operation gestalt of this invention.

[Drawing 2] It is drawing with which explanation of the migration of an agent in said agent system is presented.

[Drawing 3] It is drawing with which explanation of migration of said agent is presented.

[Drawing 4] It is drawing with which explanation of migration of said agent is presented.

[Description of Notations]

1 Agent system,

2 User terminal

3 Agent server,

4 Agent generation section,

5 Result Management Department,

6 Status-control section,

AG, AG1, AG2, AG3 Agent,

AP1, AP2, AP3 Application server

[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

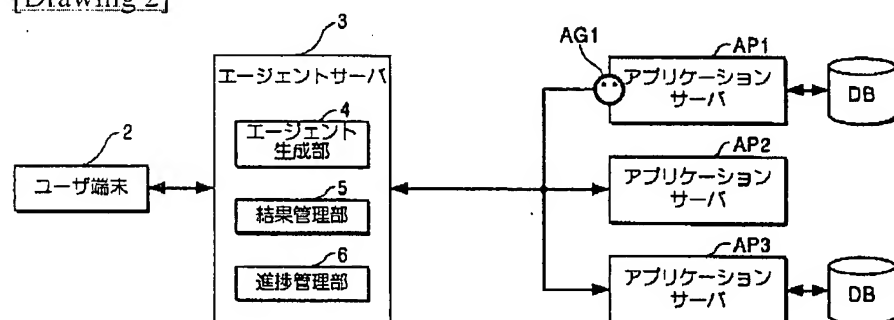
1. This document has been translated by computer. So the translation may not reflect the original precisely.

2. **** shows the word which can not be translated.

3. In the drawings, any words are not translated.

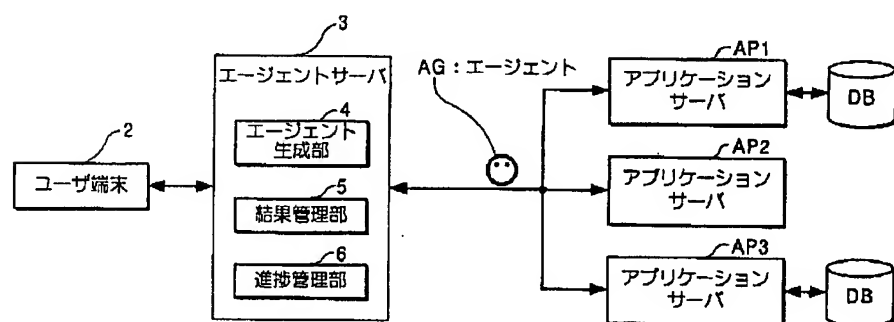
DRAWINGS

[Drawing 2]

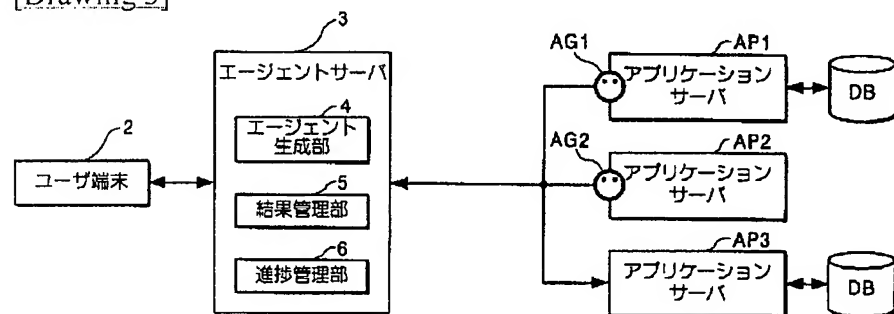


[Drawing 1]

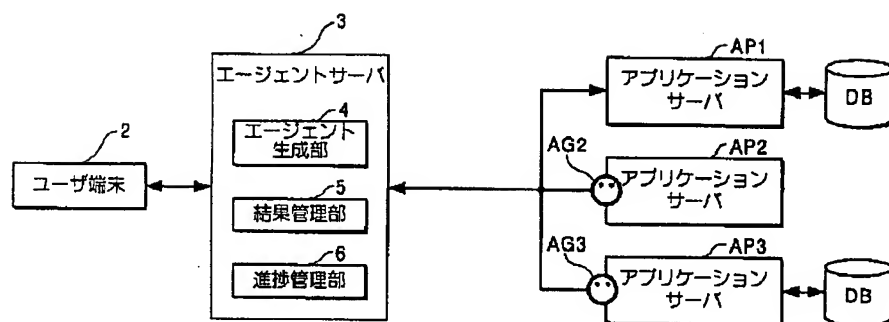
1



[Drawing 3]



[Drawing 4]



[Translation done.]

(2)

特開2001-273272

1

2

【特許請求の範囲】

【請求項1】 ユーザの要求に応じてエージェントサーバがネットワーク上を移動可能なエージェントを生成することにより、前記エージェントを介して前記ネットワークに接続された複数のアプリケーションサーバにおいて前記ユーザの要求に対応する処理を行うエージェントシステムの制御方法において、

前記エージェントは、移動先の前記アプリケーションサーバにおける処理結果の取得が完了すると、その時点における自己の複製を作成してコピーエージェントとして当該アプリケーションサーバに待機させ、

当該エージェントは、次の移動先である前記アプリケーションサーバまたは前記エージェントサーバに移動することを特徴とするエージェントシステムの制御方法。

【請求項2】 請求項1記載のエージェントシステムの制御方法において、

前記コピーエージェントが予め定められた数になった後は、前記エージェントが移動する度に現存する前記コピーエージェントのうち少なくとも最も最初に作成されたコピーエージェントを含むコピーエージェントが削除されることを特徴とするエージェントシステムの制御方法。

【請求項3】 請求項2記載のエージェントシステムの制御方法において、

前記現存するコピーエージェントのうち最新のコピーエージェントが前記削除すべきコピーエージェントを削除することを特徴とするエージェントシステムの制御方法。

【請求項4】 請求項1記載のエージェントシステムの制御方法において、

前記エージェントサーバは、前記エージェントの現在の位置及び状態を管理する進捗管理部を有し、前記進捗管理部が前記エージェントの消滅若しくは障害を検出した場合は、現存する前記コピーエージェントのいずれかが前記消滅若しくは障害を検出されたエージェントの処理を継続すべく所定の処理を行うことを特徴とするエージェントシステムの制御方法。

【請求項5】 請求項1記載のエージェントシステムの制御方法において、

前記コピーエージェントのいずれかは、前記エージェントの消滅若しくは障害を検出した場合、前記消滅若しくは障害を検出されたエージェントの処理を継続すべく所定の処理を行うことを特徴とするエージェントシステムの制御方法。

【請求項6】 請求項4または5に記載のエージェントシステムの制御方法において、

前記所定の処理は、前記消滅若しくは障害を検出されたエージェントがロックした前記アプリケーションサーバのロックを解除する処理であることを特徴とするエージェントシステムの制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、エージェントシステムの制御方法に関する。

【0002】

【従来の技術】近年、各種処理を行わせるためのプログラムであるエージェントをアプリケーションサーバに直接移動させて実行させるエージェントシステムが提案されている。このエージェントシステムは、ユーザ端末と、エージェントを生成すると共に、エージェントによって取得された情報をユーザ端末に通知するエージェントサーバと、エージェントからの要求に対応する処理を行って処理結果をエージェントに返す複数のアプリケーションサーバとがそれぞれ通信回線で接続されて構成される。

【0003】すなわち、エージェントシステムにおいては、ユーザ端末を介してユーザからの要求があると、エージェントサーバによりユーザの要求に対応する結果を取得するためのエージェントが生成され、このエージェントが先ず最初に移動すべきアプリケーションサーバに移動する。そして、エージェントは、移動したアプリケーションサーバにユーザの要求に対応する処理を行わせてその処理結果の取得を完了すると、次の移動先のエージェントサーバに移動する。このようにして、エージェントが複数のアプリケーションサーバに順次移動して、ユーザの要求に対応する結果を取得すると、エージェントサーバに戻り、取得結果をエージェントサーバに通知する。これにより、エージェントサーバは、ユーザが要求した結果をユーザ端末を介してユーザに通知することができる。これにより、エージェントシステムは、従来のクライアントサーバシステムのようにクライアント側のサーバとアプリケーションサーバとの回線を常時接続する必要がないので、少ない通信量でアプリケーションサーバから所望の情報を取得することができる。

【0004】

【発明が解決しようとする課題】ところで、このような分散アプリケーション環境のエージェントシステムにおいては、エージェントが存在するアプリケーションサーバで処理失敗等の障害が発生すると、エージェントがその障害情報を取得し、最初のアプリケーションサーバに戻るロールバックや直前の移動先に戻るロールフォワード等の処理を行って処理を継続可能にしたものがある。しかし、この方法はエージェントが障害情報を取得することが必要になるため、アプリケーションサーバのシステムダウンなどのエージェント自身が消滅するような障害が発生した場合は考慮されていなかった。このため、エージェント自身が消滅した場合は、処理が中断したり、たとえエージェントが消滅したことを検出できたとしても、結局、エージェントを新たに生成して改めて処理を行う必要があった。

(3)

特開2001-273272

3

4

【0005】そこで本発明の目的は、エージェントが消滅するような障害が発生した場合でも、障害発生時までそのエージェントが取得した情報を用いて続く処理を継続することができるエージェントシステムの制御方法を提供することを目的とする。

【0006】

【課題を解決するための手段】上記課題を解決するため、請求項1記載の構成は、ユーザの要求に応じてエージェントサーバがネットワーク上を移動可能なエージェントを生成することにより、前記エージェントを介して前記ネットワークに接続された複数のアプリケーションサーバにおいて前記ユーザの要求に対応する処理を行うエージェントシステムの制御方法において、前記エージェントは、移動先の前記アプリケーションサーバにおける処理結果の取得が完了すると、その時点における自己の複製を作成してコピーエージェントとして当該アプリケーションサーバに待機させ、当該エージェントは、次の移動先である前記アプリケーションサーバまたは前記エージェントサーバに移動することを特徴としている。

【0007】請求項2記載の構成は、請求項1記載のエージェントシステムの制御方法において、前記コピーエージェントが予め定められた数になった後は、前記エージェントが移動する度に現存する前記コピーエージェントのうち少なくとも最も最初に作成されたコピーエージェントを含むコピーエージェントが削除されることを特徴としている。

【0008】請求項3記載の構成は、請求項2記載のエージェントシステムの制御方法において、前記現存するコピーエージェントのうち最新のコピーエージェントが前記削除すべきコピーエージェントを削除することを特徴としている。

【0009】請求項4記載の構成は、請求項1記載のエージェントシステムの制御方法において、前記エージェントサーバは、前記エージェントの現在の位置及び状態を管理する進捗管理部を有し、前記進捗管理部が前記エージェントの消滅若しくは障害を検出した場合は、現存する前記コピーエージェントのいずれかが前記消滅若しくは障害を検出されたエージェントの処理を継続すべく所定の処理を行うことを特徴としている。

【0010】請求項5記載の構成は、請求項1記載のエージェントシステムの制御方法において、前記コピーエージェントのいずれかは、前記エージェントの消滅若しくは障害を検出した場合、前記消滅若しくは障害を検出されたエージェントの処理を継続すべく所定の処理を行うことを特徴としている。

【0011】請求項6記載の構成は、請求項4または5に記載のエージェントシステムの制御方法において、前記所定の処理は、前記消滅若しくは障害を検出されたエージェントがロックした前記アプリケーションサーバのロックを解除する処理であることを特徴としている。

【0012】

【発明の実施の形態】以下、適宜図面を参照しながら本発明の実施形態について説明する。

【0013】(1) 実施形態

(1-1) 実施形態の構成

図1は、本発明の実施形態に係るエージェントシステムを示すブロック図である。このエージェントシステム1は、ユーザがこのシステムを利用するための端末装置であるユーザ端末2と、ユーザ端末2からの要求に対応する結果を取得してユーザ端末2に返すエージェントサーバ3と、エージェントAGからの要求に対する処理を行って処理結果をエージェントAGに返すというアプリケーションを実行する複数のアプリケーションサーバAP1～AP3とがそれぞれ通信回線で接続されて構成される。

【0014】エージェントサーバ3は、エージェント生成部4と、結果管理部5と、進捗管理部6とから構成される。エージェント生成部4は、ユーザ端末2の要求を解析し、解析結果に応じて図示しないメモリに記憶されている所定のソースプログラムを読み出して所定のプログラムを生成し、このプログラムに移動先情報を付加して後述するエージェントAGを生成する。結果管理部5は、各エージェントAGがアプリケーションサーバAP1～AP3より取得した処理結果を管理し、ユーザ端末2にユーザの要求に対応する情報を送信する。進捗管理部6は、各エージェントAGの現在の位置や生存の状態を監視し、エージェントAG（後述するこのエージェントAGに対応するコピーエージェントを含む）が処理の継続が不可能であると判断した場合には、このエージェントAGを再び作成させて処理を継続させたり、中止処理を行う。

【0015】アプリケーションサーバAP1～AP3は、上述したように、エージェントAGからの要求に対して所定の処理を行うアプリケーションを実行するサーバであり、例えば、図1において、アプリケーションサーバAP1及びAP3は、エージェントAGの要求に応じてデータベースDBの内容を更新したり、データベースDBの中から所望のデータを検索して検索結果をエージェントAGに返すサーバである。

【0016】エージェントAGは、ユーザの要求に対する結果を取得すべくエージェントサーバ3とアプリケーションサーバAP1～AP3との間及びアプリケーションサーバAP1～AP3間を自ら移動するプログラムである。このエージェントAGは、上述したように移動先のアプリケーションサーバに所定の処理を実行させて処理結果を取得したり、現在位置や状態をアプリケーションサーバ3の進捗管理部6に通知するのみならず、トランザクション管理や、移動先のアプリケーションサーバにおける処理失敗などの障害が発生した場合にその後の動作を自立的に決定する障害時回復処理を行うことがで

(4)

特開2001-273272

5

きるプログラムである。

【0017】すなわち、エージェントAGが移動先のアプリケーションサーバ3の処理失敗などの障害を検出した場合、エージェントAGは、ロールバックやロールフォワードを行う方法の他、その障害が発生したアプリケーションサーバが実行すべき処理を処理可能な他のアプリケーションサーバに移動したり、エージェントサーバ3に移動して途中結果を通知するなどの様々な方法の中から最適な方法を自立的に選択することができる。従って、このエージェントシステム1においては、エー

10 ジェントAGが移動先のアプリケーションサーバAP1～AP3の障害を検出した場合は、エージェントAGがその障害に自立的に対処して処理を継続することができる。
【0018】また、このエージェントシステム1においては、エージェントAGがアプリケーションサーバAP1～AP3間を移動した際、そのエージェントAGの複製であるコピーエージェントが1つ前の移動先のアプリケーションサーバに待機するようになっている。すなわち、図2～図4にエージェントの移動方法を示すよう

20 に、例えば、図2に示すように、エージェントAG1が最初の移動先であるアプリケーションサーバAP1において処理結果の取得が完了すると、その処理結果を取得したエージェントAG1の複製を作成する。そして、図3に示すように、その複製がエージェントAG2として次の移動先であるアプリケーションサーバAP2に移動し、複製元がコピーエージェントAG1としてアプリケーションサーバAP1に待機する。

【0019】次に、アプリケーションサーバAP2に移動したエージェントAG2は、アプリケーションサーバAP2において処理結果の取得が完了すると、上述の場合と同様に、図4に示すように、それまでの処理結果を取得したエージェントAG2の複製がエージェントAG3として次の移動先であるアプリケーションサーバAP3に移動すると共に、コピーエージェントAG2がアプリケーションサーバAP2に待機する。このとき、アプリケーションサーバAP2に待機するコピーエージェントAG2によりアプリケーションサーバAP1に待機する

30 コピーエージェントAG1が削除され、最後に移動したエージェントの移動元のアプリケーションサーバのみに直前の処理結果を取得したコピーエージェントが待機するようになっている。

【0020】従って、エージェントAGが現在存在しているアプリケーションサーバのダウン等によってエー

6

ージェントがアプリケーションサーバの障害を検出した場合と同様に、迅速に障害に対処することができる。

【0021】これにより、このエージェントシステム1においては、エージェントが存在しているアプリケーションサーバのダウン等によってエージェント自体が消滅した場合でも、消滅したエージェントが最後に移動したアプリケーションサーバの移動元のアプリケーションサーバに存在するコピーエージェントによって処理を継続することができる。また、このコピーエージェントは、消滅したエージェントがその待機するアプリケーションサーバから移動するまでに取得した処理結果を有しているため、消滅したエージェントがすでに行った処理を重複して行う必要なく、迅速に処理を継続することができる。

【0022】なお、このエージェントシステム1において、複数のアプリケーションサーバに障害が発生してエージェント及び全てのコピーエージェントが消滅した場合は、エージェントサーバ3の造形管理部6がこれらの消滅を検出すると、エージェント生成部4により新たに生成されたエージェントAGによって処理を継続することができる。また、エージェントシステム1に障害が発生せず、エージェントサーバ3がエージェントAGを介してユーザ2の要求に対応する結果を取得できた場合は、エージェントサーバ3の制御によってコピーエージェントが削除される。

【0023】(1-2) 実施形態の効果

本実施形態に係るエージェントシステム1は、エージェントAGが移動先のアプリケーションサーバにおける処理結果の取得を完了すると、その処理結果を取得したエージェントの複製であるコピーエージェントをそのアプリケーションサーバに待機させておくことにより、次の移動先に移動したエージェントAGが消滅するような障害が発生した場合でも、障害発生時までにそのエージェントが取得した情報を用いて続く処理を継続することができる。

【0024】(2) 変形例

(2-1) 第1変形例

上述の実施形態においては、コピーエージェントをエージェントが現在存在しているアプリケーションサーバの移動元のアプリケーションサーバのみに待機させる場合について述べたが、本発明はこれに限らず、要はエージェントが現在存在しているアプリケーションサーバ以外のアプリケーションサーバにコピーエージェントを待機させればよく、2以上のアプリケーションサーバにコピーエージェントをそれぞれ待機させてもよい。この場合、複数のアプリケーションサーバに障害が発生してエージェントといずれかのコピーエージェントが消滅した場合でも、正常なアプリケーションサーバに待機するコピーエージェントのうち最後に作成されたコピーエージェントによって処理を効率よく継続することができる。

(5)

特開2001-273272

7

【0025】また、コピーエージェントの数が予め定めた数になった後は、コピーエージェントが新たに作成される度に現存するコピーエージェントのうち最も最初に作成されたコピーエージェントを削除するのが望ましいが、削除するコピーエージェントを待機するアプリケーションサーバの処理内容などに基づいて決定してもよく、また、待機させるコピーエージェントの数が動的に変わってもよい。なお、コピーエージェントを削除せずに、エージェントが移動したアプリケーションサーバの全てにコピーエージェントを待機させてもよい。

【0026】(2-2) 第2変形例

上述の実施形態においては、エージェントの消滅を検出したエージェントサーバ3の道徳管理部5が続く処理をコピーエージェントに指示する場合について述べたが、本発明はこれに限らず、エージェントとコピーエージェントの間のエージェント通信によってコピーエージェントが直接エージェントの消滅を検出してよい。

【0027】(2-3) 第3変形例

上述の実施形態においては、エージェントが消滅した場合にのみ最後に作成されたコピーエージェントが消滅したエージェントの処理を継続する場合について述べたが、本発明はこれに限らず、エージェント自体に障害が発生して処理継続不能になった場合、障害が発生したエージェントに代わって最後に作成されたコピーエージェントが続く処理を行ってもよい。

【0028】(2-4) 第4変形例

また、上述の実施形態において、あるエージェントが要求する処理を実行中のアプリケーションサーバが他のエージェントによる処理を実行してしまうことを防止するため、エージェントが処理実行中のアプリケーションサーバをロックさせるロック処理を行うエージェントシステムに本発明を適用してもよい。このエージェントシステムにおいて、アプリケーションサーバに移動したエージェントが障害により消滅した場合は、消滅したエージェントがロックしたアプリケーションサーバはロックされたままになってしまうので、このロックの解除を最後*

8

*に作成されたコピーエージェントが行ってもよい。

【0029】(2-5) 第5変形例

上述の実施形態においては、エージェントサーバに1つのユーザ端末が接続された分散アプリケーション環境のエージェントシステムに本発明を適用する場合について述べたが、本発明はこれに限らず、エージェントサーバに複数のユーザ端末が接続された分散アプリケーション環境のエージェントシステムや、エージェントサーバが複数ある分散アプリケーション環境のエージェントシステムに本発明を広く適用することができる。

【0030】

【発明の効果】上述したように本発明のエージェントシステムは、エージェントが消滅するような障害が発生した場合でも、障害が発生していないアプリケーションサーバに待機するコピーエージェントによって障害発生時までそのエージェントが取得した情報を用いて続く処理を継続することができる。

【図面の簡単な説明】

【図1】 本発明の実施形態に係るエージェントシステムのブロック図である。

【図2】 前記エージェントシステムにおけるエージェントの移動の説明に供する図である。

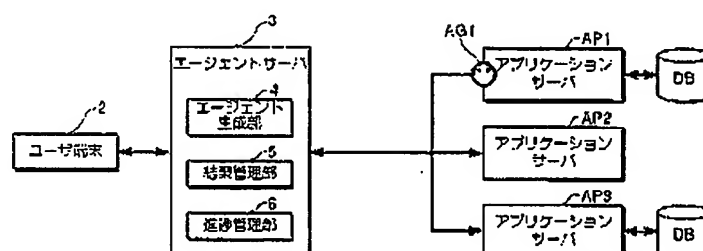
【図3】 前記エージェントの移動の説明に供する図である。

【図4】 前記エージェントの移動の説明に供する図である。

【符号の説明】

- 1……エージェントシステム、
- 2……ユーザ端末、
- 3……エージェントサーバ、
- 4……エージェント生成部、
- 5……結果管理部、
- 6……道徳管理部、
- AG、AG1、AG2、AG3……エージェント、
- AP1、AP2、AP3……アプリケーションサーバ

【図2】

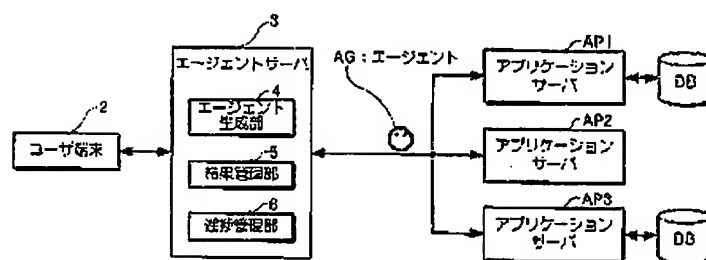


(5)

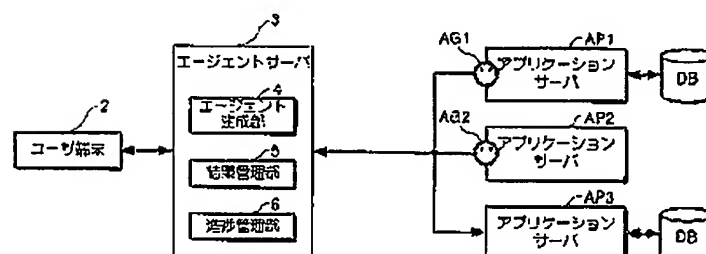
特開2001-273272

【図1】

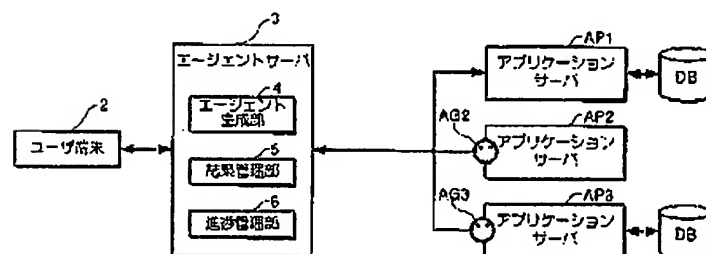
1



【図3】



【図4】



 フロントページの続き

(72)発明者 長岡 亨
 東京都港区港南一丁目9番1号 エヌ・テ
 イ・ティ・コミュニケーションウェア株式
 会社内

Fターム(参考) 5B034 BB01 CC03
 5B045 GG01 JJ44
 5B082 DA02 DE03